A Project Report on

"Hospital Management System"

SUBMITTED TO

G H RAISONI COLLEGE OF ENGINEERING MANAGEMENT

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

SUBMITTED BY-

SUYOG S. IGAVE	**TY AI-A73**
SHIVAM V. RANDIVE	**TY AI-A62**
TANAY S. MAPARE	**TY AI-A75**
SHIVAM S. SHEWALKAR	**TY AI-A60**

UNDER THE GUIDANCE OF

Prof. Vaishali Baviskar

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

G H RAISONI COLLEGE OF ENGINEERING MANAGEMENT, WAGHOLI,PUNE.

2022-23

CERTIFICATE

This is to certify that the minor project report entitles

"Hospital Management System"

Submitted By

SUYOG IGAVE	**TY AI-A73**
SHIVAM RANDIVE	**TY AI-A62**
TANAY MAPARE	**TY AI-A75**
SHIVAM SHEWALKAR	**TY AI-A60**

are the bonafide students of this institute and the work has been carried out by them under the supervision of **Prof. Vaishali Baviskar** and it is approved for the partial fulfilment of the requirement.

Prof. Vaishali Baviskar
Guide

Prof. Rachna Sable
Head of Department

Dr. R. D. Kharadkar
Campus Director
GHRCEM, Pune

ACKNOWLEDGEMENT

It gives us great pleasure and satisfaction in presenting this project report on “Hospital Management System”.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of AI Department which helped us in successfully completing our project work. Also, we would like to extend our sincere esteems to all staff in laboratory for their timely support.

We have furthermore to thank AI Department HOD Prof. R. Y. Sable and Guide Prof. Vaishali Baviskar to encourage us to go ahead and for continuous guidance. We would also like to thank our project team members who showedimmense patience and understanding throughout the project.

We would like to thank all those, who have directly or indirectly helped us for the completion of the work during this project.

1. Suyog Igave
2. Shivam Randive
3. Tanay Mapare
4. Shivam Shewalkar

TABLE OF CONTENTS

Chapter			Page No
1	**INTRODUCTION**		
	1.1	PROBLEM STATEMENT	2
	1.2	PROJECT OVERVIEW	2
	1.3	PROJECT OBJECT	2
	1.4	PROJECT SCOPE	2
2	**RELATED WORK**		
	2.1	EXISTING SYSTEM	3
3	**SYSTEM DESIGN**		
	3.1	PROPOSED SYSTEM	4
	3.2	SYSTEM DESIGN	5
	3.3	ER-DIAGRAM	6
4	**METHODOLOGY**		7
	4.1	Front-end	
	4.2	Server Side Script	
	4.3	Database	
5	**SYSTEM REQUIREMENTs**		
	5.1	HARDWARE REQUIRMENTS	8
	5.2	SOFTWARE REQUIRMENTS	
6	**RESULTS**		
	6.1	SCREEN SHOTS	9
7	**CONCLUSION**		13
8	**REFERENCES**		14

ABSTRACT

Our project Hospital Management system includes registration of patients, storing their details into the system, and also booking their appointments with doctors. Our software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. User can search availability of a doctor and the details of a patient using the id. The Hospital Management System can be entered using a username and password. It is accessible either by an administrator or receptionist. Only they can add data into the database. The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast. It is having mainly two modules. One is at Doctor Level and other one is of user I.e. of patients and doctors. The Application maintains authentication in order to access the application. To achieve this aim a database was designed one for the patient and other for the doctors which the admin can access. The complaints which are given by user will be referred by authorities

1. INTRODUCTION

This project is a web-based management system for hospitals. The project objective is to deal with challenges in traditional hospital management methods.

Our project Hospital Management system includes registration of patients, storing their details into the system, and also booking their appointments with doctors. Our software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. User can search availability of a doctor and the details of a patient using the id. The Hospital Management System can be entered using a username and password. It is accessible either by an administrator or receptionist. Only they can add data into the database. The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast..

1.1 PROBLEM STATEMENT:

In this busy world we don't have the time to wait in infamously long hospital queues. The problem is, queuing at hospital is often managed manually by administrative staff, then take a token there and then wait for our turn then ask for the doctor and the most frustrating thing - we went there by traveling a long distance and then we come to know the doctor is on leave or the doctor can't take appointments.

HMS will help us overcome all these problems because now patients can book their appointments at home, they can check whether the doctor they want to meet is available or not. Doctors can also confirm or decline appointments, this help both patient and the doctor because if the doctor declines' appointment then patient will know this in advance and patient will visit hospital only when the doctor confirms' the appointment this will save time and money of the patient. Patients can also pay the doctor's consultant fee online to save their time.

1.2 OBJECTIVES:

This software will help the company to be more efficient in registration of their patients and manage appointments, records of patients. It enables doctors and admin to view and modify appointments schedules if required. The purpose of this project is to computerize all details regarding patient details and hospital details.

1.3 SCOPE:

The system will be used as the application that serves hospitals, clinic, dispensaries or other health institutions. The intention of the system is to increase the number of patients that can be treated and managed properly. If the hospital management system is file based, management of the hospital has to put much effort on securing the files. They can be easily damaged by fire, insects and natural disasters. Also could be misplaced by losing data and information.

2. RELATED WORK

2.1 EXISTING SYSTEM/ Papers:

The current system for booking an appointment is to visit the hospital manually and wait for doctor to be free. This traditional methods has following challenges;

1. Patient must go to hospital and take appointment.
2. It is difficult to manage past records..
3. It is a time-consuming process

3. SYSTEM DESIGN

Our application contains two modules – the Doctor module and the Patient module. Our application will not only help the Doctor to preview the monthly and/or yearly data but it will also allow them to edit, add or update records. The software will also help the Doctor to monitor the transactions made by the patients and generate confirmations for the same. The Doctor will be able to manage and update information about doctors. The user module can be accessed by both the doctors and the patients. The patients will be able to apply for the appointment and make transaction for the same, and can even cancel appointments with the doctors. They can track details about the previous transactions made by them.

3.1 SYSTEM DESIGN

1. **PATIENT**

REGISTRATION DESCRIPTION –

- The new patient can register themselves and add their details like name, age , gender, blood group etc.
- The patient entry will be made in the hmdbms database.

PRE -CONDITION – The patient must be a new patient, If necessary fields left by user then prompt user to fill the necessary fields.

MAIN FLOW OF EVENTS

1. Patient selects sign up in login module.
2. A registration form get displayed
3. Patient fills the required details.

POST CONDITIONS - Patient record is added to hmdbms database.

UPDATION DESCRIPTION-

- The patient should be enabled to update his/her details and the changes should reflect in hms database.

PRE-CONDITION – The patient must be a registered patient, The patient cannot update details after treatment starts.

MAIN FLOW OF EVENTS

1. Patient logs in to the system.
2. Patient view his record
3. Patient selects update details.
4. Now patient may change the necessary fields.
5. Pop of update details.

POST CONDITION - The record of patient is updated in hms database.

APPOINTMENT DESCRIPTION

- It shows users a list of available doctors, timings, dates and enables patients to select the most suitable appointment date and doctor.
- The patient may also the cancel the appointment. PRE-CONDITION - The patient must be a registered patient.
  Patient can fix only one appointment for a particular department.

MAIN FLOW OF EVENT

1. Patient first logs in to system.
2. View his/her record.
3. Create a new appointment or cancel the appointment..

POST CONDITIONS - patient details are displayed and a new appointment is fix or a existing appointment is cancelled.
The hmdbms database is updated.

PAYMENT DESCRIPTION –

- It enables user to pay the consultant fee of Doctor online.

PRE-CONDITION –

- The patient must be a registered patient, If Patient don't wants to pay online he/she can pay by cash also.

MAIN FLOW OF EVENT

1. Patient first logs in to system.
2. View his/her record.
3. Appointment confirmed by the Doctor then go for Payment.

POST CONDITIONS –

- A Reciept will be displayed.
- he hmdbms database is updated

2. **DOCTOR DESCRIPTION-**

- The doctor view patient record/ update his details and add description of the treatment given to patient.

PRE-CONDITION –

- The doctor must be a registered doctor, System does not allow the doctor to modify the qualification, hospital managed details.

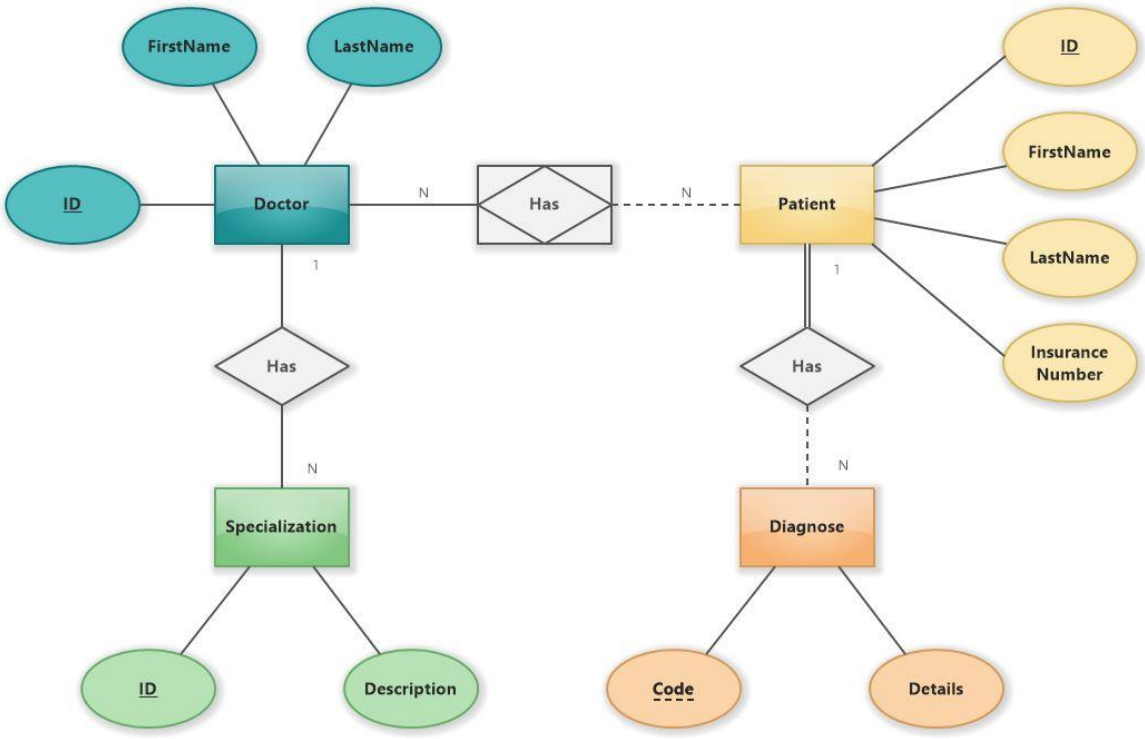
MAIN FLOW OF EVENTS

1.Doctor logs in to the system.
2. Doctor may select view patient.
2.1 Patient record is displayed with treatment history.
3. Doctor add description of patient treatment.
4. Doctor may select appointment details
4.1 Appointment Requests is displayed with schedule.
5. Doctor confirm or cancel appointment.

POST CONDITION

- The patient and doctor ‘s database are updated..

3.3 E R Diagram



4. METHODOLOGY

4.1 Front-end : HTML , CSS , JAVASCRIPT

Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.

4.2 Server Side Script : Python (Flask)

Flask is a web application framework written in Python. It is developed by **Armin Ronacher**, who leads an international group of Python enthusiasts named Pocco. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Libraries Used

```
from flask import Flask, render_template, request, session, redirect, url_for, flash
from flask_sqlalchemy import SQLAlchemy
from flask_login import UserMixin
from werkzeug.security import generate_password_hash, check_password_hash
from flask_login import login_user, logout_user, login_manager, LoginManager
from flask_login import login_required, current_user
from flask_mail import Mail
import json
```

4.3 DATABASE : MYSQL

MySQL is one of the most recognizable technologies in the modern big data ecosystem. Often called the most popular database and currently enjoying widespread, effective use regardless of industry, it's clear that anyone involved with enterprise data or general IT should at least aim for a basic familiarity of MySQL.

4.4 SOFTWARE USE :

1. Visual Studio Code
2. XAMPP Server

5. SYSTEM REQUIREMENTS

5.1 H/W Requirement

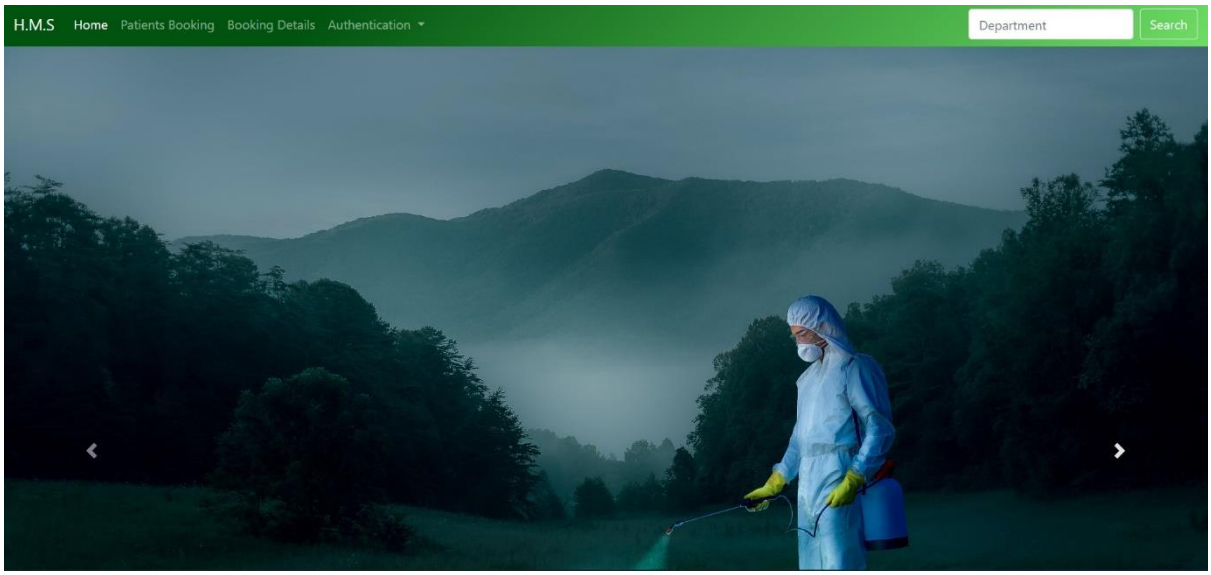
- Core i5 processor
- 2GB Ram.
- 20GB of hard disk space in terminal machines
- 1TB hard disk space in Server Machine

5.2 S/W Requirement

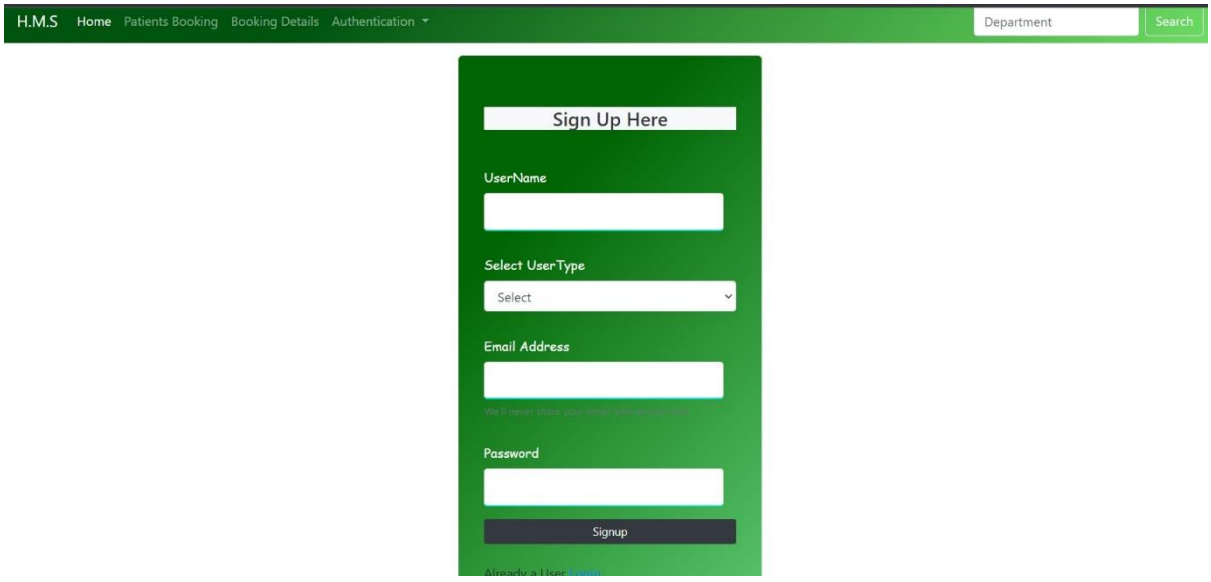
- Windows 7 or above operating system
- JRE 1.8
- Mysql serve

6. RESULTS

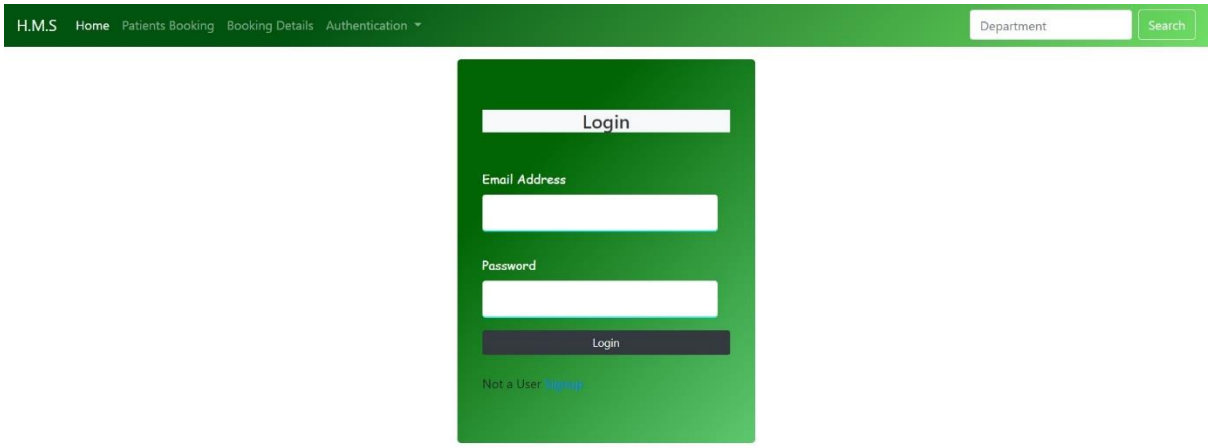
Home page :



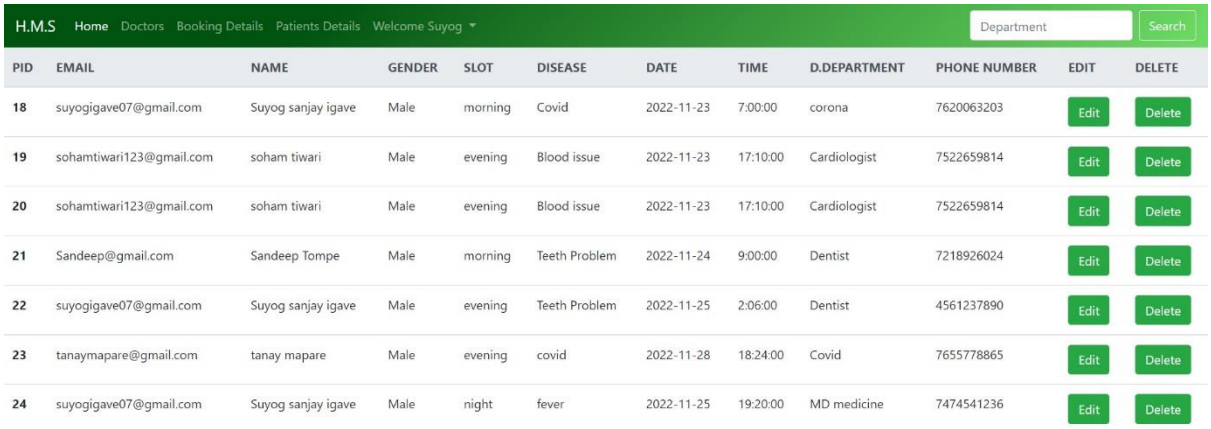
Sign Up page :



Login page :

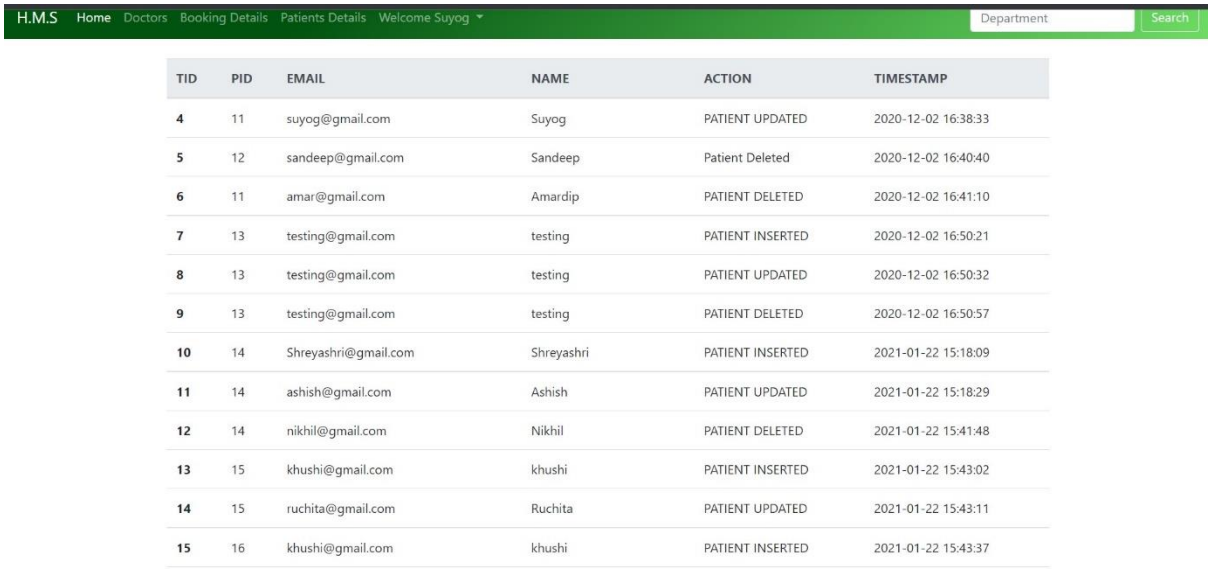


Appointment Details :



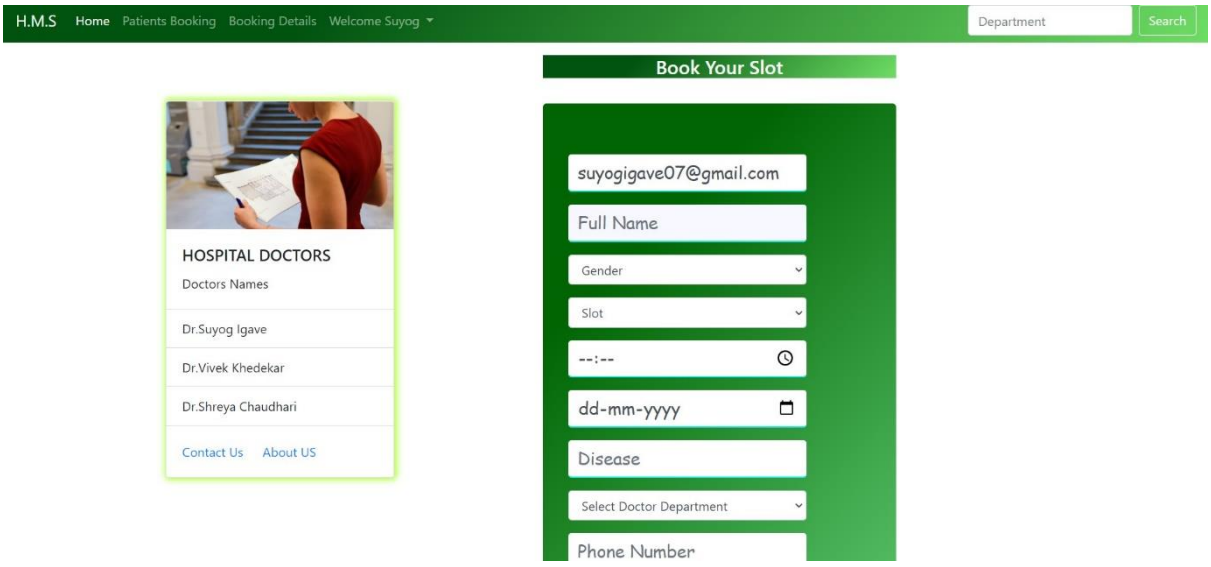
PID	EMAIL	NAME	GENDER	SLOT	DISEASE	DATE	TIME	D.DEPARTMENT	PHONE NUMBER	EDIT	DELETE
18	suyogigave07@gmail.com	Suyog sanjay igave	Male	morning	Covid	2022-11-23	7:00:00	corona	7620063203	Edit	Delete
19	sohamtiwari123@gmail.com	soham tiwari	Male	evening	Blood issue	2022-11-23	17:10:00	Cardiologist	7522659814	Edit	Delete
20	sohamtiwari123@gmail.com	soham tiwari	Male	evening	Blood issue	2022-11-23	17:10:00	Cardiologist	7522659814	Edit	Delete
21	Sandeep@gmail.com	Sandeep Tompe	Male	morning	Teeth Problem	2022-11-24	9:00:00	Dentist	7218926024	Edit	Delete
22	suyogigave07@gmail.com	Suyog sanjay igave	Male	evening	Teeth Problem	2022-11-25	2:06:00	Dentist	4561237890	Edit	Delete
23	tanaymapare@gmail.com	tanay mapare	Male	evening	covid	2022-11-28	18:24:00	Covid	7655778865	Edit	Delete
24	suyogigave07@gmail.com	Suyog sanjay igave	Male	night	fever	2022-11-25	19:20:00	MD medicine	7474541236	Edit	Delete

Patient History :



TID	PID	EMAIL	NAME	ACTION	TIMESTAMP
4	11	suyog@gmail.com	Suyog	PATIENT UPDATED	2020-12-02 16:38:33
5	12	sandeep@gmail.com	Sandeep	Patient Deleted	2020-12-02 16:40:40
6	11	amar@gmail.com	Amardip	PATIENT DELETED	2020-12-02 16:41:10
7	13	testing@gmail.com	testing	PATIENT INSERTED	2020-12-02 16:50:21
8	13	testing@gmail.com	testing	PATIENT UPDATED	2020-12-02 16:50:32
9	13	testing@gmail.com	testing	PATIENT DELETED	2020-12-02 16:50:57
10	14	Shreyashri@gmail.com	Shreyashri	PATIENT INSERTED	2021-01-22 15:18:09
11	14	ashish@gmail.com	Ashish	PATIENT UPDATED	2021-01-22 15:18:29
12	14	nikhil@gmail.com	Nikhil	PATIENT DELETED	2021-01-22 15:41:48
13	15	khushi@gmail.com	khushi	PATIENT INSERTED	2021-01-22 15:43:02
14	15	ruchita@gmail.com	Ruchita	PATIENT UPDATED	2021-01-22 15:43:11
15	16	khushi@gmail.com	khushi	PATIENT INSERTED	2021-01-22 15:43:37

Appointment Slot booking :



Booking detail(Patient view) :

7. CONCLUSION

The project was successfully completed after a lot of efforts and work hours. This project underwent number of compiling, debugging, removing errors, making it bug free, adding more facilities in Hospital Management System and interactivity making it more reliable and useful. This project focused that scheduling a project and adhering to that schedule creates a hard sense of time- management. It has also let us known that co-operative teamwork always produce effective results There are also few features which can be integrated with this system to make it more flexible.

8. REFERENCES

1. Programming Computer Vision with Python, 1st Edition, Jan Eric Solem, 2012, O' Reily
2. Python Flask library (https://pypi.org/project/Flask/)
3. https://www.w3schools.com/MySQL/default.asp
4. https://blog.templatetoaster.com/xampp-phpmyadmin/(For Backend)
5. https://www.edureka.co/blog/web-development-tutorial/